トランジスター回路の計算について

井上　　浩
上坂寿治
沢田富夫

Calculation of transistor-circuit.

In the calculation of electronic circuits with transistor, DC-solution and transient solution are observed, under the condition of symbolic branch impedance known, and some kinds of procedures are given.

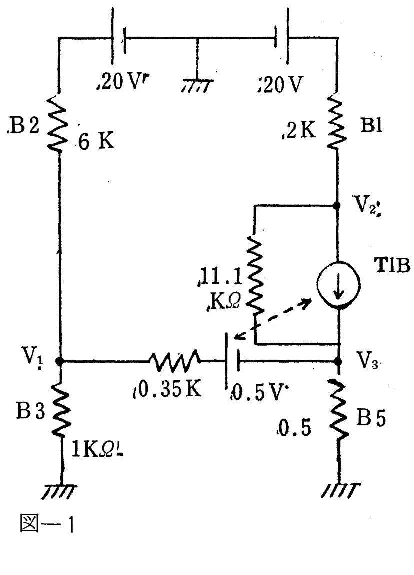
Hirosi INOUE. Tosiharu KOSAKA. Tomio SAWADA.

1．緒　言

トランジスターを含む電子回路の計算においては，ECAP，NET—１，あるいはASPECなど便利なコンパイラーが作られている。しかして機種によっては此のコンパイラーを使うことが出来ない場合がある。本文は都合によって使用出来なかったので，トランジスターを含む回路の初歩的な計算を試み，ECAPで与えられた形式より，接続行列への変換，およびインピーダンス行列式を与えて回路の過渡解を求めるプログラムについて述べてみたい。

2．接続行列への誘導

ECAPで解いてある例題について考える。図—１はトランジスターを含む直流回路であって，ECAPでは



図—１

B１	N(0,2)，R=2000，E=20
B２	N(0,1)，R=6000，E=20
B３	N(0,1)，R=1000
B４	N(1,3)，R=350，E=−0.5
B５	N(3,0)，R=500
B６	N(2,3)，R=11.1E３
T１	B(4,6)，BETA=50

図—２

図—２のような入力を与える。此の意味はB１の値が，O端子（接地）と１端子間に接続されていることを表しており，其の値は20Vの電池と2000Ωの抵抗である。T１はトランジスターで４端子と６端子の間に接続され，電流増巾率は50である。此の回路の電流又は端子と接地間の電圧を求めよと云うことになる。

各枝電流 $I_1, I_2, \cdots\cdots I_6$，各端子の電圧 V_1，V_2，V_3，としてキルヒホッフの方程式を行列式I〔１：10，１：９〕の形に直すようにする。図—２のうちで，

Nの項を　　N〔１：６，１：２〕
Rの項を　　R〔１：６〕
Eの項を　　E〔１：６〕
T１の項を　T〔１：２〕
BETAの項を　S

で表す。I〔１：10，１：９〕は最後の列にE〔１：６〕がくるようにするためで，最初の６列までは電流の項，あとの３列は電圧関係を示す。従って最初の６行，６列には対角線にR〔１：６〕が入り他は全部０が入るようI〔　〕を作らなければならない。次に電

圧に関しては7列，8列，9列は6行目まで接続された点のみ＋1または−1の値が存在する。同様に7行8行9行目は6列まで接続された点のみ＋1または−1の値がある。たとえば，B1はN(0,2)であるからV_2即ち8列に−1がありB4はN（1，3）であるから4行V_1の位置に＋1，V_3の位置に−1をおく。また電流に対して考えると各接続点における電流の総和は0であるので，第7行目には端子1における電流で矢印を考えて，第2列に−1，第3列に−1，第4列に＋1を入れることを考えるとよい。

もしトランジスターが入れば，B4の電流をβ倍してT1端子に加えるとよいので此れを実際に行なわしめると図—3のプログラムとなり，実施結果は図—4のようになる。

```
begin   real S; array I[1:9,1:10], R[1:6], E[1:6]; integer array N[1:6,1:2], T[1:2];
        integer J, K, C, D, F, G, P, Q; Format FOM[(−0.3⏨−2,('#')↑3)↑10];
        Read array (N[1:6,1:2]); Read array (R[1:6]); Read array (E[1:6]); Read array
        (T[1:2]); Read 1 (S);
        for J :=1 step 1 until 9 do
        for K :=1 step 1 until 10 do  I[J,K] :=0.0;
        for J :=1 step 1 until 6 do
        begin I[J,J] :=R[J]; I[J,10]:=E[J]; C :=N[J,1]; if C=0 then go to A1;
        I[C+6,J] :=I[J,C+6] :=1.0;
A1 :    D:=N[J,2]; if D :=0 then go to A2; I[D+6,J] :=I[J,D+6] :=−1.0;
A2 :    end;
        F:=T[1]; G :=T[2]; P :=N[G,1]+6; Q :=N[G,2]+6;
        I[P,F] :=I[P,F]+S; I[Q,F] :=I[Q,F]−S;
        for J :=1 step 1 until 9 do
        for K :=1 step 1 until 10 do Print (FOM,I[J,K]);
end ;
        N(0,2,0,1,0,1,1,3,3,0,2,3);
        R(2000.0,6000.0,1000.0,350.0,500.0,11100.0);
        E(20.0,20.0,0.0,−0.5,0.0,0.0);
        T(4,6);
```

図—3

I_1	I_2	I_3	I_4	I_5
.200$_{10}$ 4	.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50
.000$_{10}$-50	.600$_{10}$ 4	.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50
.000$_{10}$-50	.000$_{10}$-50	.100$_{10}$ 4	.000$_{10}$-50	.000$_{10}$-50
.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50	.350$_{10}$ 3	.000$_{10}$-50
.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50	.500$_{10}$ 3
.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50
.000$_{10}$-50	-.100$_{10}$ 1	-.100$_{10}$ 1	.100$_{10}$ 1	.000$_{10}$-50
-.100$_{10}$ 1	.000$_{10}$-50	.000$_{10}$-50	.500$_{10}$ 2	.000$_{10}$-50
.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50	-.510$_{10}$ 2	.100$_{10}$ 1

I_6	V_1	V_2	V_3	E
.000$_{10}$-50	.000$_{10}$-50	-.100$_{10}$ 1	.000$_{10}$-50	.200$_{10}$ 2
.000$_{10}$-50	-.100$_{10}$ 1	.000$_{10}$-50	.000$_{10}$-50	.200$_{10}$ 2
.000$_{10}$-50	-.100$_{10}$ 1	.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50
.000$_{10}$-50	.100$_{10}$ 1	.000$_{10}$-50	-.100$_{10}$ 1	-.500$_{10}$ 0
.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50	.100$_{10}$ 1	.000$_{10}$-50
.111$_{10}$ 5	.000$_{10}$-50	.100$_{10}$ 1	-.100$_{10}$ 1	.000$_{10}$-50
.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50
.100$_{10}$ 1	.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50
-.100$_{10}$ 1	.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50	.000$_{10}$-50

図—4

3. 過 渡 解

先に求められたキルヒホッフの行列式より各部の電圧または電流値を求めることが出来るが，此の行列式より過渡解を求めてみる。既にキルヒホッフの関係式が与えられており，各枝にはたかだかRとLまたはRとCのみの直列インピーダンスと考えて，此の枝のインピーダンスは$r+pL$または$r+\frac{1}{pc}$と表しうる。$r+\frac{1}{pc}=\frac{1}{p}(rp+c)$とおけば行列式の展開形式のときに留意すれば$r+pL$と全く同様な考えで差支えないため，$r+pL$の形に各枝のインピーダンスを表すものと考える。過渡解においては此の行列式が0となる根pの値を求める。此の行と列の多いときにはシグナルフローグラフ図の計算と同じであるので，次のように取扱う。即ち各素子の総ての組合せを作り，其の積を符号を考えて和を取るようにする。たとえば5行5列であれば，

$Z_{1\alpha}\ Z_{2\beta}\ Z_{3\gamma}\ Z_{4\delta}\ Z_{5\varepsilon}$

とおいて積を作り，α, β, γ, δ, ε に対しては素に素であり，同じ値ではないあらゆる組合せを作るとよいので図—5のようなプログラムを考えてもよい。

従ってECAPのように与えられた数値から過渡解を求めるためには，

a. キルヒホッフの関係式をArrayの形式に構成すること。此の時Arrayを2つに分けてpを含まない係数によるArrayとpを含む係数によるArrayの2つを組立てること。

b. 行列式の根を求めるために，前の2つArrayより展開した時pの同次のベキの係数を夫々加算してArrayの形式とし，

$H_0p^3+H_1p^2+H_2p+H_3$

のようにしH_0, H_1 …… H_3をArrayの形式にしておく。

c. 上式の根を求めて後展開定理，或はhaplace変換を行って過渡解を求める。

```
begin integer A1,A2,A3,A4,A5,I,J,K,N;
        integerarray A[1:5];
        Format F[(6)↑5];
   for A1:=1 step 1 until 5 do
   begin
   A[1]:=A1;
   for A2:=1 step 1 until 5 do
   begin
   A[2]A:=2;
   for A3:=1 step 1 until 5 do
   begin
   A[3]:=A3;
   for A4:=1 step 1 until 5 do
   bsgin
   A[4]:=A4;
   for A5:=1 step 1 until 5 do
   begin
   A[5]:=A5;
   for I:=5 step-1 until 2 do
   begin
   J:=6-I;
   switch S:=L1,L2,L3,L4;
   for K:=1 step 1 until I-1 do
   if A[I] =A[K] then
   go to S[J];
   end;
   for I:=1 step 1 until 5 do
   Print (F,A[I]);
  L1: end;
  L2: end;
  L3: end;
  L4: end;
end;
end
```

図—5

```
begin    integer NO, L, F, NN, M, J, I, M1, K, I1, I2, I3, I4, N;
         real E1, E2, X4, SUM, X2, X3, Y2, Y3, T, N1, X1, Y1;
         array X, Y, U, CONV, V[1:4], Q[1:2, 1:5], S[0:5], A[0:4], P1, P2
         [1:4, 1:5];
         integer array AA[1:3, 1:3];
Format   F1[('#')↑2, (2. 2, ('#')↑2, -1. 5₁₀-2)↑6],
         F2[('###'-1. 5₁₀-2)↑5];
Procedure ROOTPOL;
         begin integer I, J, M, N, REV, SGN;
         array H, B, C, D, E[-2:4];
         real T, P, Q, S, R1, R2, R3, DET, RD, SQRTD, EPS, ETA;
            B[-1]:=B[-2]:=C[-1]:=C[-2]:=D[-1]:=D[-2]:=E[-1]:=E;
          [-2]:=0
          N:=4;
          I:=5;
         for J:=0 step 1 until N do
           begin
             I:=I-1;
             H[J]:=A[I];
           end; SGN:=1;
ZEROTEST:
           if H[N]=0.0 then begin U[N]:=V[N]:=0.0; CONV[N]:=1.0/10.0↑F;
           N:=N-1; go to ZEROTEST end;
INIT:
         if N=0 then go to RETURN;
         REV:=1; ETA:=10.0↑F; EPS:=1.0/ETA;
         if N=1 then begin T:=-H[1]/H[0]; go to LINEAR end;
         if N=2 then begin p:=H[1]/H[0]; Q:=H[2]/H[0]; go to QADRTIC end;
         If abs (H[1]/H[0])<abs (H[N-1]/H[N]) then
REVERSE:
         begin SGN:=-SGN; M:=(N-1)÷2;
         for J:=O step 1 until M do begin S:=H[J]; H[J]:=H[N-1]; H[N-J]:=S; end;
         end;
APPROX: if abs(H[N-2])<EPS then Q:=P:=1.0 else
         begin Q:=H[N]/H[N-2]; P:=(H[N-1]-Q*H[N-3])/H[N-2]; end;
         if abs (H[N-1])<EPS then T:=1.0 else T:=-H[N]/H[N-1];
ITERATE:
         for I:=1 step 1 until L do
           begin
```

図—6

```
           forJ:=0 step 1 until N do
           begin
BAIRSTOW:  B[J]:=H[J]—P*B[J-1]-Q*B[J-2];
           C[J]:=B[J]-P*C[J-1]-Q*C[J-2];
NEWTON:    D[J]:=H[J]+T*D[J-1];
           E[J]:=D[J]+T*E[J-1];
           end;
           R1:=B[N-1]; R2:=H[N]-Q*B[N-2]; R3:=D[N];
RITEST:    if abs (R1)<EPS then go to R2TEST;
           if abs (H[N-1]/R1) <ETA then go to R3TEST;
R2TEST:    if abs (R2)<ETA V abs (H[N]/R2)>ETA then go to QADRTIC;
R3TEST:    if abs (R3)<EPS V abs (H[N]/R3)>ETA then go to LINEAR;
           C[N-1]:=-P*C[N-2]-Q*C[N-3];
           DET:=C[N-2]↑2-C[N-1]*C[N-3];
           if DET=0.0 then
           begin P:=P+1.0; Q:=Q+1.0 end
             else
               begin
                 P:=P+(B[N-1]*C[N-2]-B[N]*C[N-3])/DET;
                 Q:=Q+(-B[N-1]*C[N-1]+B[N]*C[N-2])/DET;
             end;
               if E[N-1]=0.0 then T:=T-1.0 else T:=T-D[N]/E[N-1];
           end;
           if REV<0 then begin EPS:=10.0*EPS; ETA:=ETA/10.0; end;
           REV:=-REV; go to REVERSE;
LINEAR:    if SGN<0 then T:=1.0/T;
           U[N]:=T; V[N]:=0.0; CONV[N]:=EPS; N:=N-1;
           for J:=0 step 1 until N do H[J]:=D[J];
           go to INIT;
QADRTIC:   P:=P*0.5;
           if SGN<0 then begin P:=P/Q; Q:=1.0/Q; end;
           RD:=Q-P↑2;
           if RD<0.0 then
             begin
            U[N]:=U[N-1]:=-P; SQRTD:=sqrt(RD); V[N]:=SQRTD; V[N-1]:=-SQRTD;
             end
           else
             begin
               SQRTD:=sqrt(-RD);
             if P<0.0 then U[N]:=-P+SQRTD else U[N]:=-P-SQRTD;
             U[N-1]:=Q/U[N]; V[N-1]:=V[N]:=0.0
           end;
           CONV[N]:=CONV[N-1]:=EPS;
           N:=N-2;
           for J:=0 step 1 until N do H[J]:=B[J];
```

図—6

```
            go to INIT;
RETURN:
            end;
procedure MUL;
  begin
    integer II1, II2, II3, II4;
    real SA1, SA2;
    S[5]:=S[1]:=1.0;
    for II1:=1 step 1 until 4 do
      begin
        S[1]:=S[1]*X[II1];
        S[5]:=S[5]*Y[II1];
      end;
      S[3]:=S[4]:=S[2]:=0.0;
    for II1:=1 step 1 until 4 do
      begin
        SA1:=Y[II1];
        SA2:=X[II1];
      for II2:=1 step 1 until 4 do
        if II1≒II2 then begin SA1:=SA1*X[II2];  SA2:=SA2*Y[II2]; end;
        S[2]:=S[2]+SA1;
        S[4]:=S[4]+SA2;
      end;
    for II1:=1 step 1 until 4 do
    for II2:=II1+1 step 1 until 4 do
      begin
        for II3:=1 step 1 until 4 do
          if II1≒II3 ∧ II2≒II3 then
      begin
        for II4:=II3+1 step 1 until 4 do
          if II1≒II4 ∧ II2≒II4 then begin S[3]:=S[3]+X[II1]*X[II2]*Y[II3]*Y[II4]; go to
            END;end;
       end;
    END:
      end;
        end;
    procedure ELEM;
      begin
        if I≒I1 then
          begin NO:=NO+1;  J:=AA[I2,NO];
                X[I]:=P1[I,J];  Y[I]:=P2[I,J];
          end;
        end;
      procedure PRAS;
        begin integer II;
          for II:=1 step 1 until 5 do
```

図—6

```
      Q[N,II]:=Q[N,II]+S[II];
    end;
procedure MAINAS;
   begin integer II,
      for II:=1 step 1 until 5 do
         Q[N,II]:=Q[N,II]-S[II];
      end;
comment main program;
Read 2(F,L);
Read array (P1[1:4,1:5]);
Read array (P2[1:4,1:5]);
Read array (AA[1:3,1:3]);
for I:=1,2 do
for K:=1 step 1 until 5 do
   Q[I,K]:=0.0;
for N:=1,2 do
begin
for I1:=1 step 1 until 4 do
   begin
   for I2:=1 step 1 until 3 do
      begin
         NO:=I:=0;
         K:=I1÷2*2;
 LA1:
      I:=I+1;
      ELEM;
      if I≠4 then go to LA1;
      X[I1]:=P1[I1,1];
      Y[I1]:=P2[I1,1];
      MUL;
      if K=I1 then MAINAS else PRAS;
      NO:=0;  I:=5;
 LA2:
      I:=I-1;
      ELEM;
      if I≠1 then go to LA2;
  MUL;
  if K=I1 then PRAS else MAINAS;
end;
end;
  for K:=1 step 1 until 4 do
    begin
      P1[K,1]:=P1[K,5];
      P2[K,1]:=P2[K,5];
    end;
  end;  Feed(3);
```

図—6

```
        for I:=1,2 do
          for J:=1 step 1 until 5 do
            Print (F2, Q[I,J]);
            Feed (1);
          for M:=0 step 1 until 4 do
            begin
              A[M]:=float (M)*Q[1,M+1];
              Print(F2,A[M]);
            end;  Feed(2);
pause;
    ROOTPOL;
  for I:=1 step 1 until 4 do
    begin Print (F2,U[I],V[I]);  Printout;  end;  Feed(5);
      for T:=0.0 step 0.01 until 10.0 do
        bsgin
          SUM:=Q[2,1]/Q[1,1];  X2:=A[4];
          Y2:=0.0;  X3:=Q[2,5];  Y3:=0.0;
        for K:=1  step 1 until 4 do
          begin
            X1:=U[K];  Y1:=V[K];
            if X1=0.0∧Y1=0.0   then X4:=0.0 else
          begin
          for I·=3 step -1 until 0 do
            begin
              X2:=X1*X2-Y1*Y2;
              Y2:=X1*Y2+X2*Y1;
              X2:=X2+A[I];
            end;
          for I:=4 step -1 until 1 do
            begin
              X3:=X1*X3-Y1*Y3;
              Y3:=X1*Y3+X3*Y1;
              X3:=X3+Q[2,I];
            end;
            E1:=(X3*X2+Y3*Y2)/(X2↑2+X2↑2);
            E2:=(X2*Y3-X3*Y2)/(X2↑2+Y2↑2);
            X4:=exp (X1*T)*(E1*cos(Y1*T)-E2*sin (Y1*T));
          end;
            SUM:=SUM+X4;
            Print (F1,T,SUM);
          end;
          Print out;
        end;
end
  (5,10);
  (4.0,-3.0,0.0,0.0,1.0,-3.0,15.0,-7.0,0.0,0.0,
```

図—6

```
0.0,−7.0,27.0,−13.0,0.0,0.0,0.0,−11.0,24.0,0.0);
(5.0,−4.0,0.0,0.0,0.0,−4.0,18.0,−8.0,0.0,0.0,
0.0,−8.0,30.0,−14.0,0.0,0.0,0.0,−12.0,26.0,0.0);
(2,3,4,4,2,3,3,4,2);
```

図—6

図—6は既にキルヒホッフの法則によりArrayが形成され，pの係数によるArrayとpを含まない係数のArrayとが与えられているものとして計算するプログラムの1例を示している。

Procedure Rootpoolは，p の多項式の根を求めるProcedureである。Procedure Mulは$Z_1\alpha$，$Z_2\beta$，$Z_3\gamma$，$Z_4\delta$，$Z_5\varepsilon$のあらゆる組合せを求める前の図—5とは異なる方法を用いるprocedureであり，procedure PRASおよびMAINASは此の展開式の積和の符号（例えばサラスの方法を考えてみる）を考慮してQなる場所に積和を store する procedure である。

此等のprocedureを用いて，展開定理により過渡解を求めるのがMainprogramである。もう少し複雑な電子回路も行列の形を大きくすると計算出来るが，計算時間も相当長くなり，一般性をもたせるには更に考慮しなければならない因子が多々生じてくるものと思われる。

此の計算にあたって助言および援助を得た奥田都さんおよび計算センター石黒さんに感謝の意を表す次第である。

文　献

(1) D.christiansen: Electronics Feb6. 1967
(2) A.F.Malmberg: Electronics Feb6. 1967